VLA COMPUTER MEMORANDUM #181

THE RELATIVE PERFORMANCE OF AIPS AND ISIS

Rick Perley, Phil Diamond, Chris Flatters

28 Sept. 1989

Considerable discussion has occurred over the last summer concerning the relative merits of ISIS and AIPS for data calibration and basic imaging. Various reports, invariably anecdotal, have circulated which claimed enormous advantages of ISIS, especially with respect to its speed in performing the essential steps of data listing, gain solution, and image generation. However, no quantitative study has yet been attempted.

Recently, the AIPS programmers discovered that the CONVEX AIPS code had been compiled without optimization, which had the especially damaging effect of forcing all vector operations to be scalar. It would thus appear likely that many of the reports circulating about were based on impressions gained when the AIPS code was at a distinct disadvantage.

It is obviously important to compare the running speeds of ISIS with AIPS. The results of such tests not only allow detection and correction of errors in codes, but also will highlight programs which are especially efficient, and which can then be transferred from one set to the other. From the user's point of view, it is important to know which calibration route, if any, holds especial advantages.

For these reasons, we have performed benchmarking tests. We selected a short but representative database for comparison of the basic tasks of database filling, matrix listing, gain solution and application, polarization calibration, and imaging. No attempt was made to compare flagging and editting speeds, as these are dominated by user skill levels, as well as being intrinsically subjective. Tests comparing spectral line functions were performed separately and will be reported separately.

The remaining sections of this report cover the following items:

(1) Discussion of the database and the operations which were performed. We also discuss the methods of timing which were employed.

(2) Display of the results in tabular form.

(3) A short discussion of the differences and implications.

The Database We selected a continuum database, taken from a 1.5 hour observing run on 14 September. These observations were at 3.6cm, and included 24 antennas. (Twenty-seven antennas were present for the inital 5 minutes, but three were then transferred to a separate subarray for VLB observing.) The data were of excellent quality, and no editing was required. Despite the short duration, polarization calibration was feasible, since the observer thoughtfully observed a strong calibrator throughout the run, giving sufficient parallactic angle coverage for an excellent solution.

The tests were performed after ensuring all other users were off the system. We monitored both wall clock time (through use of the UNIX 'date' command), and computer CPU time. This latter quantity seems a little slippery to quantify, so we attempted numerous methods. For AIPS tasks, we recorded the CPU and elapsed times, as given by each task upon completion. Note that the CPU times reported by AIPS are the sum of system plus user time. Some ISIS tasks report the CPU usage (broken down into 'user' and 'system' times), and we recorded these. For all ISIS tasks, we monitored CPU usage through the UNIX 'ps' and 'w' commands. The results of these two always closely tracked (within 5 seconds) the accounting information given by the program itself. It proved not possible to monitor the AIPS tasks in the same way, since the shed task disappears from the system upon completion. However, by monitoring the task performance as it executed, we are confident that the statistics given upon completion by the AIPS task can be compared to those determined for the comparable ISIS task within a few seconds.

It is interesting to compare the database sizes. On the Dec-10, the two data bases (for the 'AC', and 'BD' IFs) together took 13,480 blocks, or 6.90 Mbytes. On ISIS, after filling with the same PASSFLAG parameters, the database took 7.80 Mbytes. Comparing ISIS with AIPS required filling with PASSFLAG = BOTH, after which the ISIS database took 8.34 Mbytes, the AIPS database 8.28 Mbytes. Thus, the use of data compression in AIPS has made the databases in the two systems the same size to better than 1%.

The various tasks are described below, with the ISIS/AIPS names given as indicated.

1. DBFIL/FILLM The data were located approximately halfway down the tape. We monitored both the wall clock time and CPU time to space down the tape to the data, and to actually read the data into the file. We ran with PASSFLAG = BOTH, to ensure the same data were filled, since AIPS has a more liberal interpretation of what constitutes bad data. It was not possible to separate CPU times in AIPS for these two operations, so only the sum is given in the table.
2. LISTER/LISTR We produced a matrix listing of the data, with ampscalar averaging over the entire scan length for all sources and scans. We printed both the averages and the rms's. The test was run twice, the first without applying the gains, and the second with gains applied in order to assess the effects on performance of gain application. (Since the data were uncalibrated, the actual listings were identical.)
3. ANTSOL/CALIB The next step was to produce the calibration parameters. The fluxes were first entered manually (using SETJY/SETJY), from values determined earlier. (No accounting was attempted here for reasons the authors hope are obvious.) We used ampscalar averaging, solving for 24 antennas, with UVLIMITS appropriate for 3C48 and 3C138. The same reference antenna (#4) was chosen to allow detailed comparison of solutions and polarizations. Listings of the solutions were produced, and these timings were separated from the solution times.
4. GTBCAL/CLCAL We used 2-point interpolation.
5. POLCAL/PCAL The database contained two calibrators which were tracked throughout – one for calibration of the source, and other specifically intended for polarization calibration. We used both sources, with the same reference antenna in both programs.
6. MAKMAP/HORUS Finally, we produced an image. Two tests were done, the first being a 512^2 image, the second a 128^2 image.

The Timing Results. The table at the end of this report displays the results of the tests. All times are in seconds. Both the CPU and elapsed times are given, as are comments pertinent to the test performed. Note that the AIPS CPU time for the data loading test is not broken down into 'move time' vs. 'load time'.

Discussion The table shows that AIPS is markedly faster than ISIS in most of the areas tested. The difference is most marked in imaging, where the ratio exceeds a factor of two. There is no great difference if Filling data, although we note that ISIS is slower in spacing down the tape, but is considerably faster in actually loading the data. It might interest some to note the tape-to-disk data transfer rates are approximately 65 and 30 Kbytes/sec for ISIS and AIPS, respectively. We do not know the reason for these differences in tape spacing and loading, but an explanation has been suggested and is being tested.

For matrix listings, AIPS has a clear advantage – nearly a factor of two. The advantage reverses for generating the solution. Apparently the cause of AIPS' relative slowness is known and will be shortly addressed. The test should be repeated after any changes are made to the algorithm. The listing of the gain solutions is similar for both systems, and in any event, takes very little time compared to the generation of solutions. The same can be said of applying the gains, which is considerably faster in ISIS, but takes only a very short time in either system. Calculation of polarization is similarly quicker in AIPS, although the factor is not large. The large difference in imaging is very significant, because in general, one calibrates the data only once, but commonly makes images many times. Overall, the results surprised the testers. We had assumed, along with everyone else, that code especially written for the CONVEX would have definite advantages in speed over the much more general purpose AIPS. But rumour and assumption are here clearly deceiving, and the results of these tests clearly show that AIPS is to be preferred for data calibration and imaging, when speed and efficiency are important.

Three additional, and very important issues relating to deciding which software package should be employed by the users (both in-house and visitors) are the questions of ease of use, reliability, and responsiveness. With regard to the first, it is certainly true that ISIS has the advantage of familiarity, while at the same time, the generality of the AIPS package and its formidable list of adverbs for the basic calibration tasks will deter use. Two comments are appropriate: (1) The VLA-specific RUN file, provided by Bill Cotton, has largely removed the tremendous burden of sorting through the adverbs in CALIB and CLCAL to find the ones relevant to VLA data. (2) The user-unfriendly gain listings provided by LISTR will shortly be rewritten to provide the relevant information without disturbing the eye. We feel that AIPS calibration will soon be as easy as ISIS calibration.

With regard to the second, the 'rise of ISIS' over the Summer was probably due more to the tremendously disturbed state of AIPS than to anything else. We cannot expect users to use programs which break in different ways on consecutive days. Stability and reliability are centrally important. Again, we are confident that these problems are behind us, and a stable AIPS is nearly here.

The last point is related to the second - having quick response to problems builds confidence. The recent structural changes to the AIPS group will, we expect, provide the needed level of support.

RESULTS OF TIMING TESTS

Task	ISIS Elapsed Time	ISIS CPU Time	AIPS Elapsed Time	AIPS CPU Time	Comments
DBFILL/FILLM	420 + 120	200 + 65	300 +270	245	Spacing + Filling
LISTER/LISTR	82	64	45	40	No calibration, 1 IF
LISTER/LISTR	150	130	80	72	With calibration, 1 IF
ANTSOL/CALIB	60	42	110	97	Solutions
ditto	14	10	19	10	Listings
GTBCAL/CLCAL	12	5	19	14	
POLCAL/PCAL	100	80	75	58	
MAKMAP/HORUS	260	250	128	140	512^2 Image
ditto	253	245	109	105	128^2 Image